Finding Optimal Control Policy in Probabilistic Boolean Networks with Hard Constraints by Using Integer Programming and Dynamic Programming

Presented by CHEN Xi
Joint work with Prof. Akutsu, Dr. Tamura and Dr. Ching
Introduction

Problems
  - Problem 1: BN control.
  - Problem 2: Finding the optimal path for PBN.
  - Problem 3: Minimizing the maximum terminal cost for PBN.

Algorithms.
  - Integer linear programming with hard constraints for BNs and PBNs.
  - Dynamic programming with hard constraints for PBNs.

Numerical Examples.

Concluding Remarks.
Control problem of genetic regulatory networks

Modeling genetic regulatory networks:

- Bayesian networks.
- Multivariate Markov chain model.
- Boolean networks (BNs)
- Probabilistic Boolean networks (PBNs)
Introduction

Boolean Networks

Nodes (genes):
\[ V = \{v_1, v_2, \ldots, v_n\}, v_i \in \{0, 1\}. \]

Boolean functions:
\[ F = \{f_1, f_2, \ldots, f_n\}. \]

Some Definitions:
- Set of input nodes: \( \text{IN}(v_i) = \{v_{i_1}, \ldots, v_{i_k}\} \).
- Indegree of \( v_i \): the number of input nodes of \( f_i(v_{i_1}, \ldots, v_{i_k}) \).
- Maximum indegree of a BN: \( K \).
Rules of the regulatory interactions:

\[ v_i(t + 1) = f_i(v_{i_1}(t), \ldots, v_{i_k}(t)), \quad i = 1, 2, \ldots, n. \]  

Global States (States):

\[ v(t) = (v_1(t), v_2(t), \ldots, v_n(t))^T \]

Set of States (Set S):

\[ S = \{(v_1, v_2, \ldots, v_n)^T : v_i \in \{0, 1\}\} \]
Decimal Representation of States

\[ z(t) = 1 + \sum_{i=1}^{n} 2^{n-i} v_i(t), \quad z(t) \in [1, 2^n] \]

Binary States (v(t)):

\[ i.e. \{00, 01, 10, 11\} \]

Decimal States (z(t)):

\[ i.e. \{1, 2, 3, 4\} \]
### An Example of BN

Here we give a 2-gene BN with $K = 2$. The truth table is shown as follows:

<table>
<thead>
<tr>
<th>$z(t)$</th>
<th>$v_1(t)$</th>
<th>$v_2(t)$</th>
<th>$f_1 = \overline{v}_2(t)$</th>
<th>$f_2 = v_1(t) \land v_2(t)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

$\lor$: Logical OR; $\land$: Logical AND; $\overline{\cdot}$: Logical NOT.
According to the above truth table, we can get the following transition probability matrix:

\[
A = \begin{pmatrix}
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 \\
1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0
\end{pmatrix}.
\]

Attractor: \(\{v(t) = (1, 0)\}\) or \(\{z(t) = 3\}\).
Probabilistic Boolean Networks

Nodes:

\[ V = \{v_1, v_2, \ldots, v_n\}, \; v_i \in \{0, 1\}. \]

Boolean functions:

\[ \{f_1^{(i)}, f_2^{(i)}, \ldots, f_{l(i)}^{(i)}\}, \text{ for } v_i, i = 1, \ldots, n. \]

Probability of choosing \( f_j^{(i)} : c_j^{(i)}. \)
PBN – composed of $N$ BNs

The $j$th BN with the Boolean functions:

$$(f_{j_1}^{(1)}, f_{j_2}^{(2)}, \ldots, f_{j_n}^{(n)}), \quad 1 \leq j_i \leq l(i).$$

where $j = j_1 + \sum_{i=2}^{n} ((j_i - 1)(\prod_{k=1}^{i-1} l(k)))$.

Probability of choosing the $j$th BN:

$$q_j = q_{j_1j_2\ldots j_n} = \prod_{i=1}^{n} c_{j_i}^{(i)}.$$ 

There are at most

$$N = \prod_{i=1}^{n} l(i).$$

different possible realizations (BNs).
Then the transition probability

\[
P \{ v(t+1) = a \mid v(t) = b \} = \sum_{j=1}^{N} P \{ v(t+1) = a \mid v(t) = b, \text{the } j\text{th BN is selected} \} \cdot q_j
\]

where \( \mathcal{I} \) is the set of BNs of which the transition probability from state \( b \) to state \( a \) is 1.
The transition probability matrix:

$$A = \{a_{ij}\}, i, j = 1, 2, \ldots, 2^n.$$ 

where $a_{ij}$ is the transition probability from state $i$ to $j$. 
The transition probability matrix can be written as

$$A = \sum_{j=1}^{N} q_j A_j.$$  \hspace{1cm} (4)
Figure: Example of a PBN. Dynamics of the PBN is well described by the state transition probability and the transition diagram.
Transition matrices of BNs constituting the PBN:

\[ A_1 = \begin{pmatrix}
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1
\end{pmatrix}, \quad A_2 = \begin{pmatrix}
1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 \\
0 & 1 & 0 & 1
\end{pmatrix}, \quad A_3 = \begin{pmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 \\
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 1
\end{pmatrix}, \quad A_4 = \begin{pmatrix}
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 \\
0 & 0 & 1 & 1
\end{pmatrix}. \]

The selecting probabilities:

\[ q = (0.1, 0.2, 0.3, 0.4). \]
The transition probability matrix of PBN:

\[
A = \begin{pmatrix}
0.2 & 0.5 & 0 & 0 \\
0 & 0 & 0.6 & 0 \\
0.8 & 0 & 0 & 0 \\
0 & 0.5 & 0.4 & 1
\end{pmatrix}
\]

\[
A = 0.1A_1 + 0.2A_2 + 0.3A_3 + 0.4A_4.
\]
BN CONTROL (Akutsu, 2007)

- Internal Nodes (usual nodes): \( \{v_1, v_2, \ldots, v_n\} \).
- External Nodes (control nodes): \( \{v_{n+1}, v_{n+2}, \ldots, v_{n+m}\} \).
- Boolean Functions:
  \[ v_i(t + 1) = f_i(v_{i_1}(t), \ldots, v_{i_k}(t)), \quad i = 1, 2, \ldots, n. \]
- Internal State: \( \mathbf{v}(t) = [v_1(t), v_2(t), \ldots, v_n(t)] \).
- Control Input: \( \mathbf{u}(t) = [v_{n+1}(t), v_{n+2}(t), \ldots, v_{n+m}(t)] \).
Problem 1: BN CONTROL with Hard Constraints

Definition 1: Suppose an initial state of the network is $v^0$ and the desired state of the network is $v^M$, find a control sequence $\langle u(0), u(1), \ldots, u(M - 1) \rangle$ such that $v(0) = v^0$ and $v(M) = v^M$, and the maximum number of controls applied to the network during the finite time period $M$ is $H$ where $H < M$. 
Problem 1: BN CONTROL with Hard Constraints

Figure: Example of BN CONTROL.
Problem 2: Finding the optimal path with hard constraints

- Initial State: $v^0$.
- Desired State: $v^M$.
- Control Sequence: $\langle u(0), u(1), \ldots, u(M-1) \rangle$.
- BN Sequence: $\{j_0, j_1, \ldots, j_{M-1}\}$.
- Probability of a path: $\prod_{k=0}^{M-1} q_{j_k}$. 
Problem 2: Finding the optimal path with hard constraints

Definition 2: Suppose an initial state of the network is $v^0$ and the desired state of the network is $v^M$, find a control sequence $\langle u(0), \ldots, u(M) \rangle$ such that the probability of the path with the initial state $v^0$ and the terminal (desired) state $v^M$ is maximized, and the maximum number of controls applied to the network during the finite time period $M$ is $H$ where $H < M$. 
Problem 3: Minimizing the maximum cost

- **Internal State:** \( z_k = 1 + \sum_{i=1}^{n} 2^{n-i} v_i \).
- **Control Input:** \( u_k = 1 + \sum_{i=1}^{m} 2^{m-i} v_{n+i} \).
- **Terminal Cost:** \( C_M(z_M) \).
- **Maximum cost:** \( C_k(z_k) \). (the maximum cost of which, beginning from \( z_k \) at time step \( k \), the network can reach at the terminal time step.)
Problem 3: Minimizing the maximum cost

Definition 3: Given the terminal cost $C_M(z_M)$ for each of states $z_M \in \{1, 2, \ldots, 2^n\}$ at time step $M$, by applying external control, minimize the maximum cost $C_0(z_0)$ beginning from the given initial state $z_0$, and the maximum number of controls applied to the network is $H$ where $H < M$. 
ILP with hard constraints for BN CONTROL

Let $x_{i,t}$ represent the Boolean node $\nu_i(t)$. Logical formulation for Boolean functions (Akutsu, 2007):

$$f_i(x_{i_1,t}, \ldots, x_{i_k,t}) = \bigvee_{b_{i_1} \ldots b_{i_k} \in \{0,1\}^k} f_i(b_{i_1}, \ldots, b_{i_k}) \land \sigma_{b_1}(x_{i_1,t}) \land \cdots \land \sigma_{b_k}(x_{i_k,t}) \quad (5)$$

where

$$\sigma_b(x) = \begin{cases} x, & \text{if } b = 1. \\ \bar{x}, & \text{otherwise.} \end{cases} \quad (6)$$

$\lor$: Logical OR; $\land$: Logical AND; $\bar{\cdot}$: Logical NOT.
Also, we define $\tau_b(x)$ as

$$\tau_b(x) = \begin{cases} x, & \text{if } b = 1. \\ 1 - x, & \text{otherwise.} \end{cases} \quad (7)$$

and

$$x_{i,t+1,b_1\ldots b_k} = f_i(b_1, \ldots, b_k) \land \tau_{b_1}(x_{i_1,t}) \land \cdots \land \tau_{b_k}(x_{i_k,t})$$

Then equation (5) is equivalent to:

$$x_{i,t+1} = \bigvee_{b_1\ldots b_k \in \{0,1\}^k} x_{i,t+1,b_1\ldots b_k}.$$ 

Control node variable: $h_{i,t} \in \{0,1\}$. (If $h_{i,t} = 1$, we say the node $i$ changes its value at time step $t$.)
ILP Formulation for BN CONTROL

(Part 4-1) Maximize

\[
\sum_{i=1}^{N} x_{i,M}
\]

S.T.

\[
x_{i,t+1,b_1 \ldots b_k} \geq \sum_{j \in \{1,2,\ldots,k\}} \tau_{b_j}(x_{i,t}) - (k - 1)
\]

\[
x_{i,t+1,b_1 \ldots b_k} \leq \frac{1}{k} \sum_{j \in \{1,2,\ldots,k\}} \tau_{b_j}(x_{i,t})
\]

for all \( i \in \{1, 2, \ldots, n\}, \ t \in \{1, 2, \ldots, M - 1\} \) and \( b_{i_1} \ldots b_{i_k} \in \{0, 1\}^k \) such that \( f_i(b_{i_1}, \ldots, b_{i_k}) = 1 \).
ILP Formulation for BN CONTROL

(Part 4-2)

\[ X_{i,t+1,b_{i_1}...b_{i_k}} = 0 \]

for all \( i \in \{1, 2, \ldots, n\} \), \( t \in \{1, 2, \ldots, M - 1\} \), and \( b_{i_1} \ldots b_{i_k} \in \{0, 1\}^k \) such that \( f_i(b_{i_1}, \ldots, b_{i_k}) = 0 \).
ILP Formulation for BN CONTROL

(Part 4-3)

\[ x_{i,t+1} \leq \sum_{b_{i_1} \ldots b_{i_k} \in \{0,1\}^k} x_{i,t+1,b_{i_1} \ldots b_{i_k}} \] (11)

\[ x_{i,t+1} \geq \frac{1}{2^k} \sum_{b_{i_1} \ldots b_{i_k} \in \{0,1\}^k} x_{i,t+1,b_{i_1} \ldots b_{i_k}} \] (12)

for all \( i \in \{1, 2, \ldots, n\} \) and \( t \in \{0, 2, \ldots, M - 1\} \).

\[ x_{i,0} = v_i^0, \quad x_{i,M} = v_i^M. \]
ILP Formulation for BN CONTROL

(Part 4-4)

\[ x_{i,t} - x_{i,t+1} \leq h_{i,t}, \quad x_{i,t+1} - x_{i,t} \leq h_{i,t} \]  \hspace{1cm} (13)

for all \( i \in \{n+1, \ldots, n+m\} \) and \( t \in \{0, \ldots, M-1\} \).

\[ \sum_{t=0}^{M-1} \sum_{i=n+1}^{n+m} h_{i,t} \leq H \]  \hspace{1cm} (14)

for all \( i \in \{1, 2, \ldots, n\} \) and \( t \in \{1, 2, \ldots, M\} \).
ILP Formulation for Finding the Optimal Path

Selection Variable: $y_{r,t}$. (If $y_{r,t} = 1$, we say the $r$th BN is selected at $t$-time step.)

Selection Probabilities for BNs: \{\(q_1, q_2, \ldots, q_N\)\}.
ILP Formulation for Finding the Optimal Path

Objective Function:

Maximize

\[
\sum_{t=0}^{M-1} \sum_{r=1}^{R} - \log(q_r) \cdot y_{r,t}
\]
ILP Formulation for Finding the Optimal Path

For the constraints, we revise Equation (8) to (10) as follows:

\[ x_{i,t+1,b_1\ldots b_k} \geq \sum_{j \in \{1,2,\ldots,k\}} \tau_{b_j}(x_{i_{jr},t}) - (k - 1) + y_{r,t} - 1 \quad (15) \]

\[ x_{i,t+1,b_1\ldots b_k} \leq \frac{1}{k} \sum_{j \in \{1,2,\ldots,k\}} \tau_{b_j}(x_{i_{jr},t}) - y_{r,t} + 1 \quad (16) \]

for all \( i \in \{1,2,\ldots,n\} \), \( t \in \{1,2,\ldots,M-1\} \), \( b_1 \ldots b_k \in \{0,1\}^k \)
and \( r \in \{1,2,\ldots,R\} \) such that \( f_{i,r}(b_1,\ldots,b_k) = 1 \).

\[ x_{i,t+1,b_1\ldots b_k} \leq 1 - y_{r,t} \quad (17) \]

for all \( i \in \{1,2,\ldots,n\} \), \( t \in \{1,2,\ldots,M-1\} \), \( b_1 \ldots b_k \in \{0,1\}^k \)
and \( r \in \{1,2,\ldots,R\} \) such that \( f_{i,r}(b_1,\ldots,b_k) = 0 \).
ILP Formulation for Finding the Optimal Path

Then we also add the following constraints:

\[
\sum_{r=1}^{R} y_{r,t} = 1, \text{ for } t = 1, 2, \ldots, M - 1, \text{ and } y_{r,t} \in \{0, 1\}.
\]

for all \( t \in \{1, 2, \ldots, M\} \) and \( r \in \{1, 2, \ldots, R\} \).
Dynamic Programming for Minimizing the Maximum Cost

- Minimized Maximum Cost: $J_k(z_k, h_k)$.
- Control Function: $u(z_k, h_k)$.
- Set of Possible States: $F(z_k, u_k)$.
- Objective Function: $\min_{h_0 \in \{0, \ldots, H\}} J(z_0, h_0)$. 
Dynamic Programming for Minimizing the Maximum Cost

(Part 1) **Step 0:** Set $k = M$; $J(z_M, h_M) = C_M(z_M)$ for all $h_M = \{0, \ldots, H\}$.

**Step 1:** $k := k - 1$.

**Step 2:** For any $z_k \in \{1, \ldots, 2^n\}$ and $h_k \in \{0, \ldots, H\}$, compute

$$J(z_k, h_k) = \min_{u_k \in \{1, \ldots, 2^m\}} \max_{z_{k+1} \in F(z_k, u_k)} \begin{cases} J(z_{k+1}, h_k - 1), & \text{if } u_k \neq u(z_{k+1}, h_k - 1) \\ J(z_{k+1}, h_k), & \text{if } u_k = u(z_{k+1}, h_k). \end{cases}$$
Dynamic Programming for Minimizing the Maximum Cost

(Part 2) and

\[
    u(z_k, h_k) = \underset{u_k \in \{1, \ldots, 2^m\}}{\text{argmin}} \left\{ \begin{array}{ll}
    \max_{z_{k+1} \in F(z_k, u_k)} J(z_{k+1}, h_k - 1), & \text{if } u_k \neq u(z_{k+1}, h_k - 1). \\
    \max_{z_{k+1} \in F(z_k, u_k)} J(z_{k+1}, h_k), & \text{if } u_k = u(z_{k+1}, h_k). 
    \end{array} \right.
\]

**Step 3:** If \( k > 0 \), go back to step 1; Otherwise, stop.
Experiment 1: Comparison of ILP and DP for the optimal path problem

- 8 internal nodes: $\{v_1, v_2, \ldots, v_8\}$.
- 2 Control Nodes: $\{u_1, u_2\}$.
- Selecting Probabilities: $\mathbf{q} = (0.1, 0.1, 0.3, 0.05, 0.05, 0.2, 0.15, 0.05)$.
- Time Period: $M = 5$.
- Maximum number of Controls: $H = 3$.
- Initial State: 000...0.
- Desired State: 11...1.
- Boolean functions: Randomly generated.
Experiments

Experiment 1: Comparison of ILP and DP for the optimal path problem

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>Average time</th>
</tr>
</thead>
<tbody>
<tr>
<td>ILP method</td>
<td>0.38</td>
<td>4.22</td>
<td>3.66</td>
<td>0.69</td>
<td>6.54</td>
<td>3.10</td>
</tr>
<tr>
<td>DP method</td>
<td>20.81</td>
<td>5.93</td>
<td>5.75</td>
<td>7.77</td>
<td>9.53</td>
<td>9.96</td>
</tr>
</tbody>
</table>

Table: The results of Experiment 1
Experiment 2: Minimizing the maximum cost

- 8 internal nodes: \( \{v_1, v_2, \ldots, v_8\} \).
- 2 Control Nodes: \( \{u_1, u_2\} \).
- Selecting Probabilities: \( \mathbf{q} = (0.1, 0.2, 0.3, 0.4) \).
- Time Period: \( M = 3 \).
- Maximum number of Controls: \( H = 2 \).
- Initial State: 00\ldots0.
- Terminal Cost: \( C_M(z_M) = z_M \). (State 1: most desired; State 256: most undesired.)
- Boolean functions: Randomly generated.
Experiment 2: Minimizing the maximum cost

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum cost without control</td>
<td>245</td>
<td>252</td>
<td>249</td>
<td>244</td>
<td>239</td>
</tr>
<tr>
<td>Minimized maximum cost</td>
<td>193</td>
<td>194</td>
<td>177</td>
<td>173</td>
<td>163</td>
</tr>
</tbody>
</table>

Table: The results of Experiment 2
Experiment 3: On realistic networks

Figure: Structure of 10-gene WNT5A network.
Experiments

**Experiment 3: On realistic networks**

- 2 Control Nodes: RET-1 and HADHB.
- Selecting Probabilities:
  \[ q = (0.0011491, 0.2165696, 0.2018161, 0.0740315, 0.2018161, 0.0740315, 0.0587344, 0.0304030, 0.1348942, 0.2824019) \]
- Time Period: \( M = 10 \).
- Maximum number of Controls: \( H = 3 \).
- Initial State: 00...0.
- Desired State: 11...1.
- Terminal Cost: \( C_M(z_M) = z_M \)
- Boolean functions: Randomly generated.
Experiments

Experiment 3: On realistic networks

- Probability of Optimal Path: $1.2306 \times 10^{-8}$.
- External Control: 3 times.
- Minimized Maximum Cost: 172. (Without Control: 256)
For finding an optimal path for PBN, the ILP-based method is faster than the DP-based method.

ILP cannot be effectively applied to minimization of the maximum or average cost for PBN.